

**AMENDMENTS TO THE CLAIMS**

1. (Currently Amended) A method for verifying computer code having a plurality of lines generated by a code generating module from a model file of a system including a plurality of functions generated by a model module, the method comprising:

processing the model file, by a verification module, to determine values, inputs, outputs, function type, and syntax for each of the plurality of functions;

~~determining~~ generating expected code for the generated computer code based on the determined values, inputs, outputs, function type, and syntax for the generated computer code;

comparing each line of the generated computer code and the expected code to determine if the generated computer code includes correct values, correct inputs, correct outputs, correct functions, and correct syntax; and

transmitting an error message if ~~one or more lines~~ a line of the generated computer code does not include a correct value, a correct input, a correct output, a correct function, or a correct syntax based on the comparison.

2. (Currently Amended) The method of claim 1 further, comprising the steps of:

comparing each line of the generated computer code and the expected code to determine if the generated computer code is missing a line of code; and

transmitting the error message ~~if the~~ if the generated computer code is missing the line of code.

3. (Currently Amended) The method of claim 1, further comprising the steps of:  
comparing each line of the generated computer code and the expected code to determine if the generated computer code includes an extraneous line of code; and  
transmitting the error message if the [[the]] generated computer code includes the extraneous line of code.
4. (Currently Amended) The method of claim 1, further comprising the steps of:  
comparing each line of the generated computer code and the expected code to determine if the [[the]] generated computer code is in a logical order; and  
transmitting the error message if the generated computer code is not in the logical order.
5. (Previously Presented) The method of claim 1, further comprising the steps of:  
comparing a first header information section in the generated computer code and second header information section in the expected code to determine if the first header information section matches the second header information section; and  
transmitting the error message if the first header information section does not match the second header information section.
6. (Previously Presented) The method of claim 1, further comprising the steps of:  
comparing a first declared variable section in the generated computer code and a second declared variable section in the expected code to determine if the first declared variable section matches the second declared variable section; and  
transmitting the error message if the first declared variable section does not match the second declared variable section.

7. (Currently Amended) A computer-readable storage medium containing a set of instructions for verifying a generated computer code having a plurality of lines from a code generating module, the generated computer code automatically generated from a model file of a system having a plurality of functions and created by a model module, the set of instructions comprising:

- code that reads in the model file;
- code that determines values, inputs, outputs, function type, and syntax for each of the plurality of functions in the generated computer code;
- code that generates an expected computer code based on the determined values, inputs, outputs, function type, and syntax;
- code that reads in the generated computer code;
- code that compares each line in the generated computer code and the expected code to determine if the generated computer code includes the determined values, inputs, outputs, function type, and syntax in the expected computer code; and
- code that transmits an error message if ~~one or more lines~~ a first line in the generated computer code does not include a determined value, a determined input, a determined output, a determined function, or a determined syntax based on the comparison.

8. (Currently Amended) The medium of claim 7, wherein the set of instructions further comprises:

- code that compares each line in the generated computer code and the expected code; and
- code that transmits the error message if ~~one of the plurality of lines~~ the first line does not include the determined value, the determined input, the determined output, the determined function, the determined syntax, or combinations thereof.

9. (Currently Amended) The medium of claim 7, wherein the set of instructions further comprises:

code that compares each line in the plurality of lines and the expected code to determine if the plurality of lines includes a second line of code that is not determined in the expected code; and

code that transmits the error message if the generated computer code includes ~~any~~ the second line of code not determined in the expected computer code.

10. (Previously Presented) The medium of claim 7, wherein the set of instructions further comprises:

code that compares each line in the plurality of lines and the expected code to determine if the plurality of lines are in a correct logical order; and

code that transmits the error message if the plurality of lines are not in the correct logical order.

11. (Previously Presented) The medium of claim 7, wherein the set of instructions further comprises:

code that compares a first header information section in the generated computer code and a second header information section in the expected code to determine if the first header information section matches the second header information section; and

code that transmits the error message if the first header information section does not match the second header information section.

12. (Currently Amended) A system for verifying the contents of a generated computer code having a plurality of lines generated by a code generating module from a model file including a plurality of functions generated by a model module, comprising:

a processor operable to:

process the model file to determine values, inputs, outputs, function type, and syntax for each of the plurality of functions,

~~determine~~ generate expected computer code for the generated computer code based on the determined values, inputs, outputs, functions type, and syntax for the generated computer code,

compare each line in the generated computer code with the expected computer code to determine if the generated computer code includes correct values, correct inputs, correct outputs, correct functions, and correct syntax, and

transmit an error message if ~~one or more lines~~ a line in the generated computer code does not include a correct value, a correct input, a correct output, a correct function, or a correct syntax based on the comparison; and

a display configured to display the error message, the display coupled to the processor.

13. (Previously Presented) The system of claim 12, wherein the error message indicates if ~~each~~ a line is missing in the generated computer code.

14. (Previously Presented) The system of claim 12, wherein the error message indicates if a line of the generated computer code has any additional content.

15. (Currently Amended) The system of claim 12, wherein the processor is operable to compare each line in the expected code to the generated computer code to determine if the plurality of lines are in an expected form, and transmit the error message if one or more of the lines of code in the plurality of lines do not match the expected form.

16. (Currently Amended) The system of claim 12, wherein the processor is operable to compare each line in the expected code to the generated computer code to determine if the ~~plurality of lines~~ expected code includes ~~one or more lines~~ a line of extraneous code.

17. (Currently Amended) The system of claim 12, wherein the processor is operable to compare each line in the expected code to the generated computer code to determine if each line is in a logical order, and transmit the error message if any line in the plurality of lines is not in logical order.

18. (Previously Presented) The system of claim 12, wherein the processor is operable to compare a first header information section in the generated computer code to a second header information section in the expected computer code to determine if the first header information section matches the second header information section, and transmit the error message if the first header information section does not match the second header information section.

19-20. (Canceled)

21. (Previously Presented) The method of claim 1, further comprising the steps of:  
comparing each line in the generated computer code and the expected computer code to determine if the plurality of lines are complete; and  
transmitting an error message if the plurality of lines are incomplete.

22. (Previously Presented) The medium of claim 7, wherein the set of instructions further comprises:

code that compares each line in the generated computer code to the expected computer code to determine if the generated computer code is complete; and

code that transmits the error message if the generated computer code is incomplete.